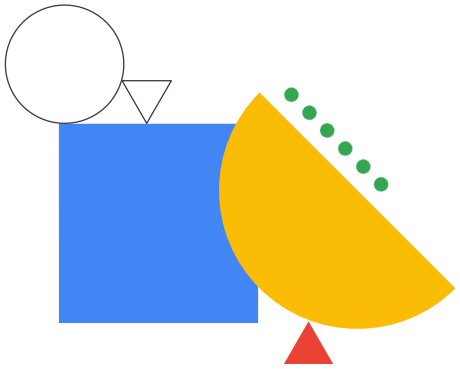
Google Cloud

Resource Management



In this module, we will cover Resource Management. Resources in Google Cloud are billable, so managing them means controlling cost. There are several methods in place for controlling access to the resources, and there are quotas that limit consumption.

In most cases, the default quotas can be raised on request, but having them in place provides a checkpoint or a chance to make sure that this really is a resource you intend to consume in greater quantity.

Agenda

01 Resource Manager

02 Quotas

03 Labels

04 Billing

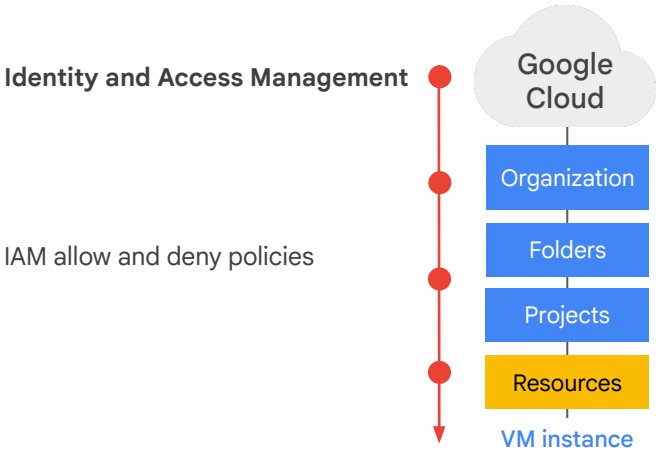
Lab: Examining Billing Data with BigQuery

In this module, we will build on what we learned in the IAM module. First, I will provide an overview of the Resource Manager. Then, we will go into quotas, labels, and names. Next, we will cover billing to help you set budgets and alerts. To complete your learning experience, you will get to examine billing data with BigQuery in a lab.

Resource Manager

Let’s get started with an overview of Resource Manager.

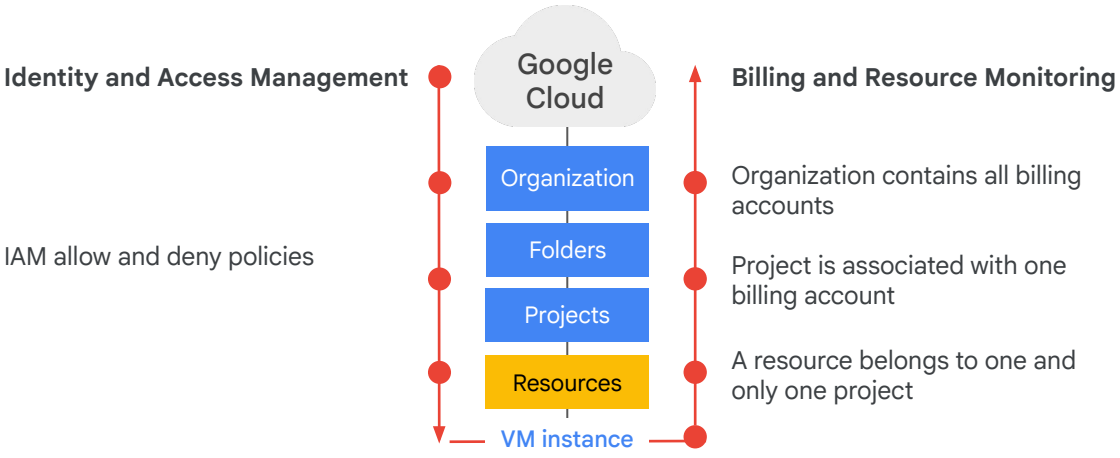
Resource Manager lets you hierarchically manage resources



The resource manager lets you hierarchically manage resources by project, folder, and organization. This should sound familiar because we covered it in the IAM module. Let me refresh your memory:

Policies contain a set of roles and members, and policies are set on resources. These resources inherit policies from their parent, as we can see on the left. Therefore, resource policies are a union of parent and resource if an IAM allow policy is associated. However, if an IAM deny policy is associated with the resource, then the policy can prevent certain principals from using certain permissions, regardless of the roles they're granted.

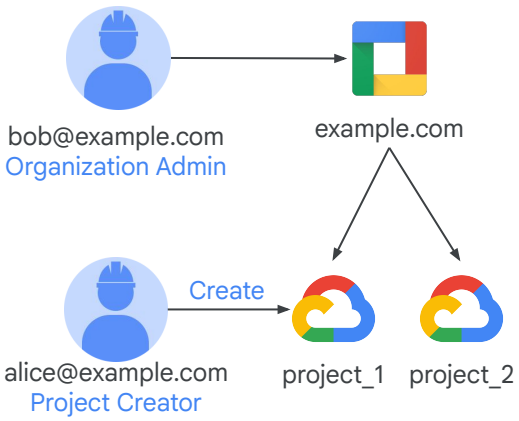
Resource Manager lets you hierarchically manage resources



Although IAM policies are inherited top-to-bottom, billing is accumulated from the bottom up, as we can see on the right. Resource consumption is measured in quantities, like rate of use or time, number of items, or feature use. Because a resource belongs to only one project, a project accumulates the consumption of all its resources.

Each project is associated with one billing account, which means that an organization contains all billing accounts. Let's explore organizations, projects, and resources more.

Organization node is root node for Google Cloud resources



Just to reiterate, an organization node is the root node for all Google Cloud resources. This diagram shows an example where we have an individual, Bob, who is in control of the organizational domain through the Organization Admin role. Bob has delegated privileges and access to the individual projects to Alice by making her a Project Creator.

Project accumulates the consumption of all its resources

- Track resource and quota usage
  - Enable billing
  - Manage permissions and credentials
  - Enable services and APIs
- Projects use three identifying attributes:
  - Project Name
  - Project Number
  - Project ID, also known as Application ID

Because a project accumulates the consumption of all its resources, it can be used to track resources and quota usage. Specifically, projects let you enable billing, manage permissions and credentials, and enable services and APIs.

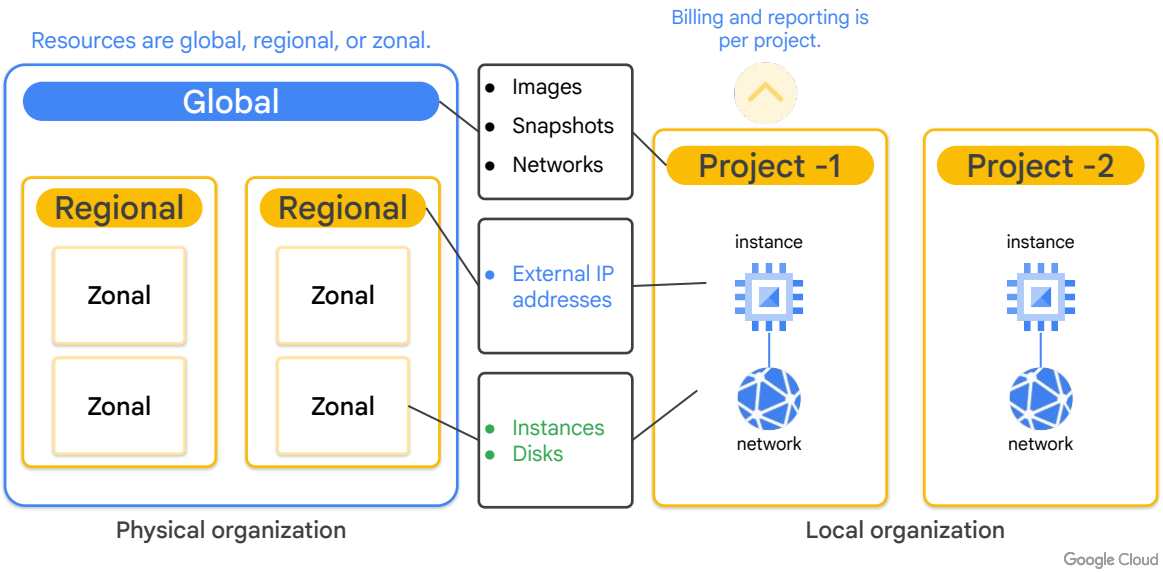
To interact with Google Cloud resources, you must provide the identifying project information for every request.

A project can be identified by:

- The project name, which is a human-readable way to identify your projects, but it isn't used by any Google APIs.
- There is also the project number, which is automatically generated by the server and assigned to your project.
- And there is the project ID, which is a unique ID that is generated from your project name.

You can find these three identifying attributes on the dashboard of your Google Cloud console or by querying the Resource Manager API.

Resource hierarchy



Finally, let's talk about the resource hierarchy. From a physical organization standpoint, resources are categorized as global, regional, or zonal.

Let's look at some examples:

- Images, snapshots, and networks are global resources;
- External IP addresses are regional resources;
- and instances and disks are zonal resources.

However, regardless of the type, each resource is organized into a project. This enables each project to have its own billing and reporting.

Quotas

Now that we know that a project accumulates the consumption of all its resources, let’s talk about quotas.

All resources are subject to project quotas or limits

- How many resources you can create per project
  - *15 VPC networks/project*
- How quickly you can make API requests in a project: rate limits
  - *5 admin actions/second (Spanner)*
- How many resources you can create per region
  - *24 CPUs region/project*

Increase: Quotas page in the Google Cloud console or a support ticket

All resources in Google Cloud are subject to project quotas or limits. These typically fall into one of the three categories shown here:

- How many resources you can create per project. For example, you can only have 15 VPC networks per project.
- How quickly you can make API requests in a project or rate limits. For example, by default, you can only make 5 administrative actions per second per project when using the Spanner API.
- There are also regional quotas. For example, by default, you can only have 24 CPUs per region.

Given these quotas, you may be wondering, how do I spin up one of those 96-core VMs?

As your use of Google Cloud expands over time, your quotas may increase accordingly. If you expect a notable upcoming increase in usage, you can proactively request quota adjustments from the Quotas page in the Google Cloud console. This page will also display your current quotas.

If quotas can be changed, why do they exist?

Why use project quotas?

- Prevent runaway consumption in case of an error or malicious attack
- Prevent billing spikes or surprises
- Forces sizing consideration and periodic review

Project quotas prevent runaway consumption in case of an error or malicious attack. For example, imagine you accidentally create 100 instead of 10 Compute Engine instances using the gcloud command line.

Quotas also prevent billing spikes or surprises. Quotas are related to billing, but we will go through how to set up budgets and alerts later, which will really help you manage billing.

Finally, quotas force sizing consideration and periodic review. For example, do you really need a 96-core instance, or can you go with a smaller and cheaper alternative?

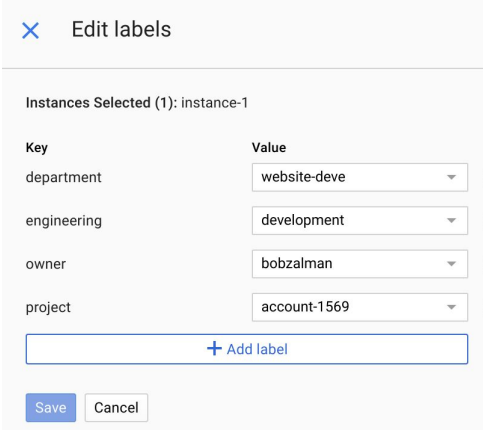
It is also important to mention that quotas are the maximum amount of resources you can create for that resource type *as long as those resources are available*. Quotas do not guarantee that resources will be available at all times. For example, if a region is out of local SSDs, you cannot create local SSDs in that region, even if you still had quota for local SSDs.

Labels

Projects and folders provide levels of segregation for resources, but what if you want more granularity? That's where labels come in.

Labels are a utility for organizing Google Cloud resources

- Attached to resources: VM, disk, snapshot, image
  - Google Cloud console, `gcloud`, or API
- *Example uses of labels*:
  - Inventory
  - Filter resources
  - In scripts
    - Help analyze costs
    - Run bulk operations



Labels are a utility for organizing Google Cloud resources. Labels are key-value pairs that you can attach to your resources, like VMs, disks, snapshots and images. You can create and manage labels using the Google Cloud console, gcloud, or the Resource Manager API, and each resource can have up to 64 labels.

For example, you could create a label to define the environment of your virtual machines. Then you define the label for each of your instances as either production or test. Using this label, you could search and list all your production resources for inventory purposes.

Labels can also be used in scripts to help analyze costs or to run bulk operations on multiple resources. The screenshot shows an example of 4 labels that are created on an instance.

Use labels for ...

- Team or Cost Center

  `team:marketing`
  `team:research`

- Components

  `component:redis`
  `component:frontend`

- Environment or stage

  `environment:prod`
  `environment:test`

- Owner or contact

  `owner:gaurav`
  `contact:opm`

- State

  `state:inuse`
  `state:readyfordeletion`

Let's go over some examples of what to use labels for:

- I recommend adding labels based on team or cost center to distinguish instances owned by different teams. You can use this type of label for cost accounting or budgeting. For example, team:marketing and team:research.
- You can also use labels to distinguish components. For example, component:redis, and component:frontend.
- Again, you can label based on environment or stage.
- You should also consider using labels to define an owner or a primary contact for a resource. For example, owner:gaurav, contact:opm.
- Or add labels to your resources to define their state. For example, state:inuse, state:readyfordeletion

Comparing labels and network tags

Labels are a way to organize resources across Google Cloud

- Disks, image, snapshots...
- User-defined strings in `key-value` format
- Propagated through billing

Network tags are applied to instances only

- User-defined strings
- Tags are primarily used for networking (applying firewall rules)

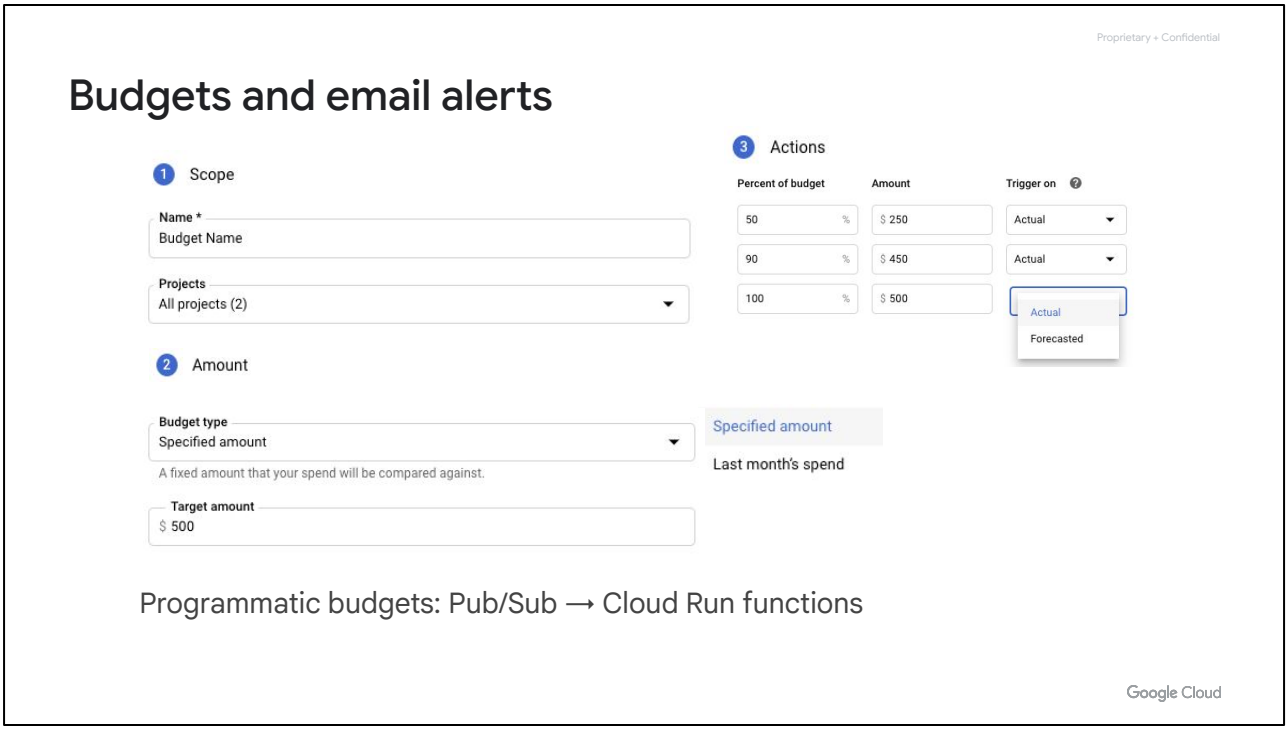
It's important to not confuse labels with network tags.

- Labels, we just learned, are user-defined strings in key-value format that are used to organize resources, and they can propagate through billing.
- Network tags, on the other hand, are user-defined strings that are applied to instances only and are mainly used for networking, such as applying firewall rules and custom static routes.

.
For more information about using labels, refer to the documentation [https://cloud.google.com/resource-manager/docs/using-labels]

Billing

Because the consumption of all resources under a project accumulates into one billing account, let’s talk billing.



To help with project planning and controlling costs, you can set a budget. Setting a budget lets you track how your spend is growing toward that amount. This screenshot shows the budget creation interface:

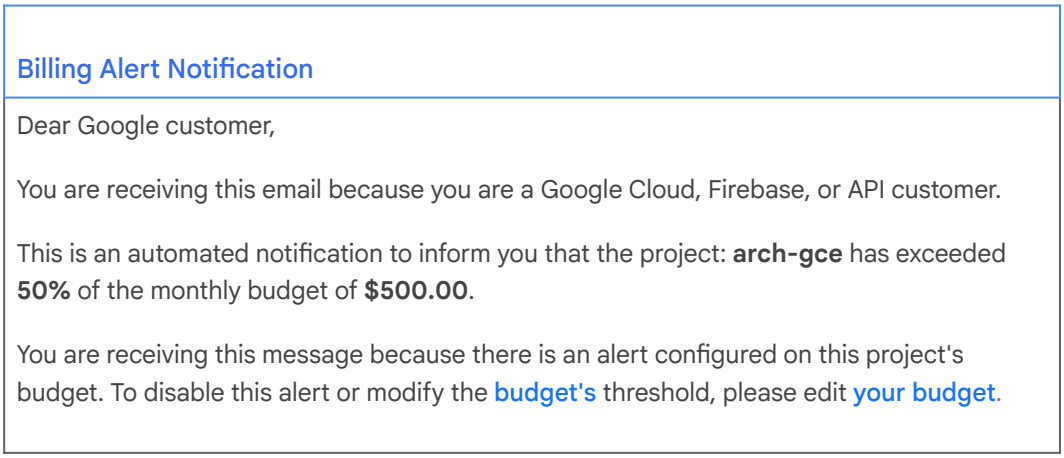
- First, you set a budget name and specify which project this budget applies to.
- Then, you can set the budget at a specific amount or match it to the previous month's spend.
- After you determine your budget amount, you can set the budget alerts. These alerts send emails to Billing Admins after spend exceeds a percent of the budget or a specified amount.

In our case, it would send an email when spending reaches 50%, 90%, and 100% of the budget amount. You can even choose to send an alert when the spend is forecasted to exceed the percent of the budget amount by the end of the budget period.

In addition to receiving an email, you can use Pub/Sub notifications to programmatically receive spend updates about this budget. You could even create a Cloud Run function that listens to the Pub/Sub topic to automate cost management.

Additional reading: [Examples of automated cost control responses | Cloud Billing]

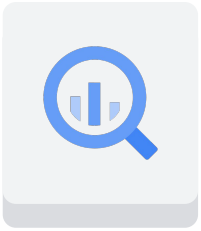
Example budget alert email



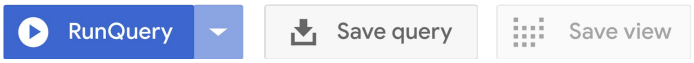
Here is an example of an email notification. The email contains the project name, the percent of the budget that was exceeded, and the budget amount.

Labels can help you optimize Google Cloud spend

```
1  SELECT
2    TO_JSON_STRING(labels)as labels,
3    sum(cost) as cost
4  FROM `project.dataset.table`
5  GROUP BY labels;
```



BigQuery



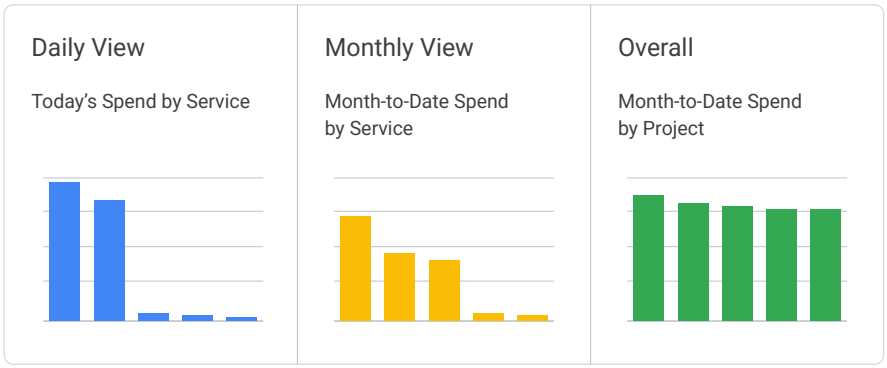
Another way to help optimize your Google Cloud spend is to use labels. For example, you could label VM instances that are spread across different regions. Maybe these instances are sending most of their traffic to a different continent, which could incur higher costs. In that case, you might consider relocating some of those instances or using a caching service like Cloud CDN to cache content closer to your users, which reduces your networking spend.

I recommend labeling all your resources and exporting your billing data to BigQuery to analyze your spend. BigQuery is Google's scalable, fully managed enterprise data warehouse with SQL and fast response times.

Creating a query is as simple as shown in this screenshot, which you will explore in the upcoming lab.

Visualize Google Cloud spend with Looker Studio

Billing Dashboard

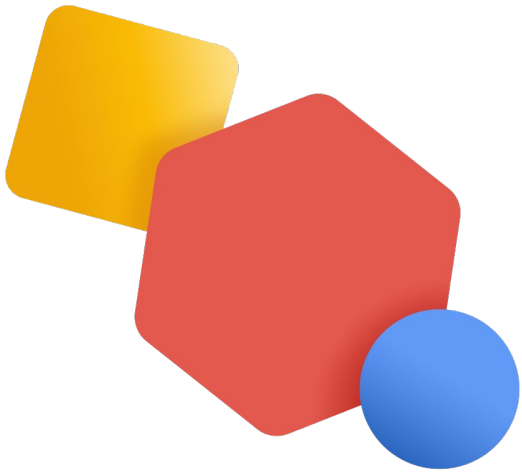


Looker Studio

You can even visualize spend over time with Looker Studio. Looker Studio turns your data into informative dashboards and reports that are easy to read, easy to share, and fully customizable. For example, you can slice and dice your billing reports using your labels.

Lab Intro

Examining Billing Data with BigQuery

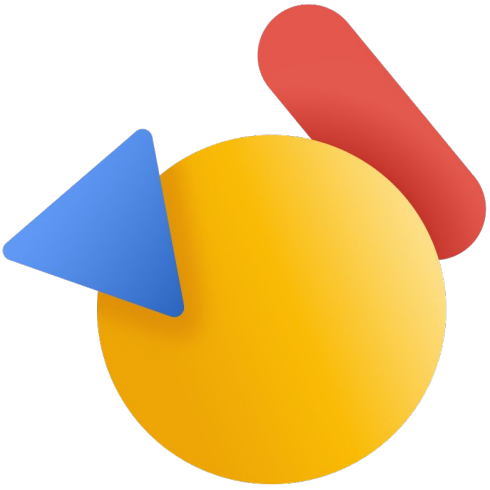


Let’s examine billing data with BigQuery.

In this lab, you will sign in to BigQuery and create a dataset. In this dataset, you will create a table by importing billing data that is stored in a Cloud Storage bucket. Next, you will run simple queries on the imported data, and then you will run more complex queries on a larger dataset.

Review

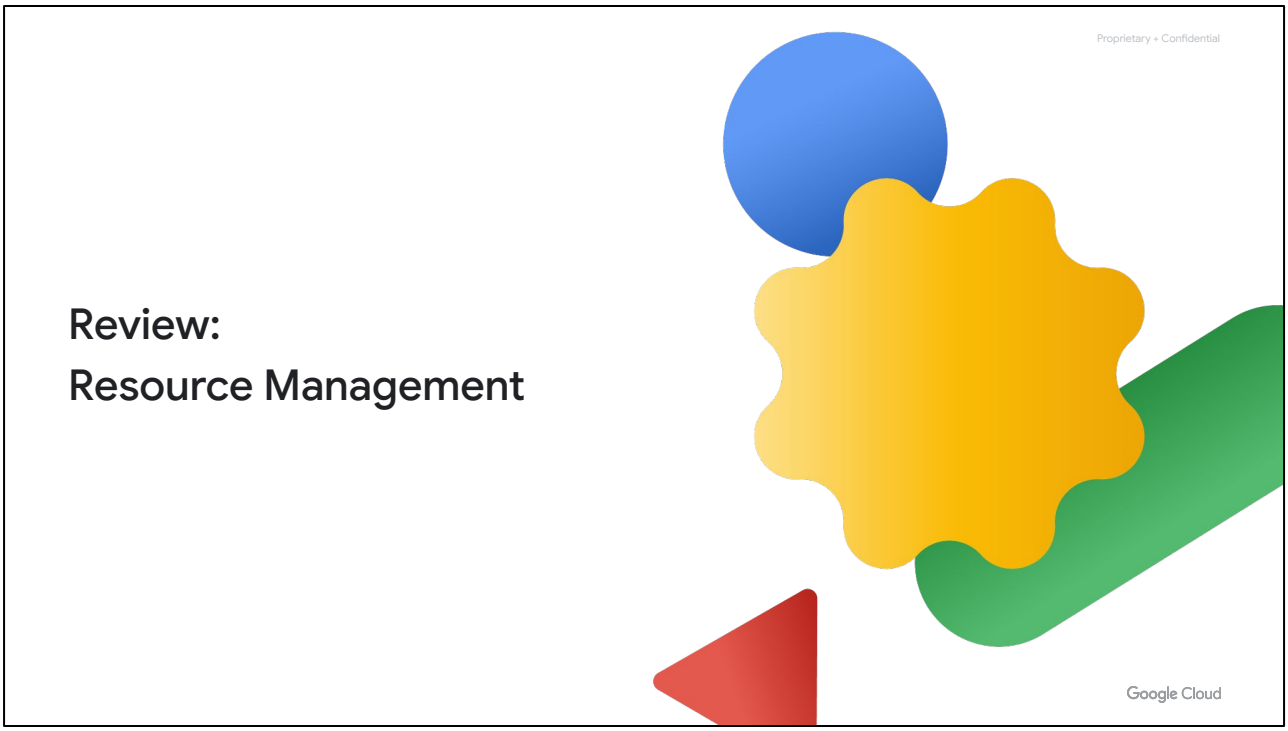
Examining Billing Data with BigQuery



In this lab, you imported billing data into BigQuery that had been exported as a CSV file. You first ran a simple query on that data.

Next, you accessed a shared dataset containing more than 22,000 records of billing information. You then ran a variety of queries on that data to explore how you can use BigQuery to gain insight into your resources' billing consumption.

If you use BigQuery on a regular basis, you'll start to develop your own queries for searching out where resources are being consumed in your application. You can also monitor changes in resource consumption over time. This kind of analysis is an input to capacity planning and can help you determine how to scale up your application to meet growth or scale down your application for efficiency.



In this module, we covered the Cloud Resource Manager and went into quotas, labels, and billing. Then we examined billing data with BigQuery in a lab.

Reporting is an important part of resource management. You can generate reports to track consumption and to establish accountability. A key principle in Google Cloud is transparency, and that means it's straightforward to access and process consumption data, as you observed in this module.